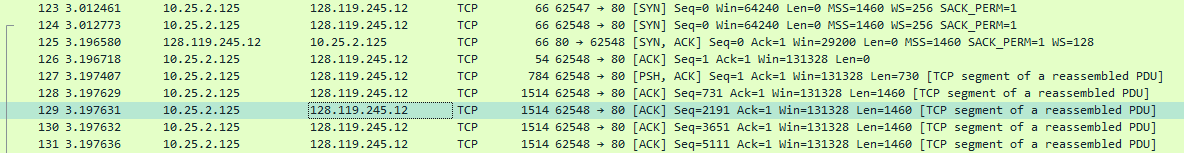
Figure #1



1. What is the IP address and TCP port number used by the client computer (source) that is transferring the file to gaia.cs.umass.edu? To answer this question, it’s probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the “details of the selected packet header window” (refer to Figure 2 in the “Getting Started with Wireshark” Lab if you’re uncertain about the Wireshark windows.

Answer: from figure #1, source IP address is 10.25.2.125 and port 62548

2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?

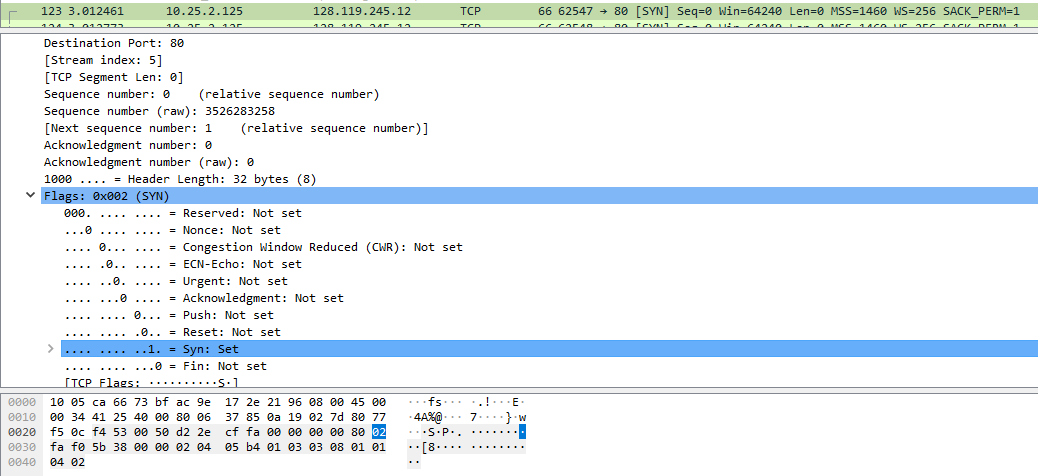
Answer: from figure #1, IP address is 128.119.245.12 and port 80

3. What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu?

Answer: from figure #1, source IP address is 10.25.2.125 and port 62548

4. What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu? What is it in the segment that identifies the segment as a SYN segment?

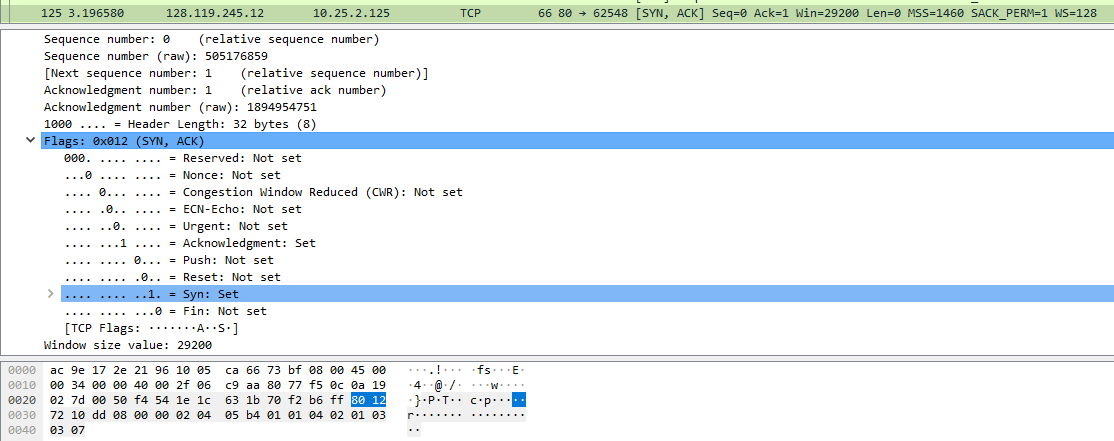
Figure #2



Answer: from figure #2, Sequence number 0, Flags: 0x002 (SYN)

5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?

Figure #3

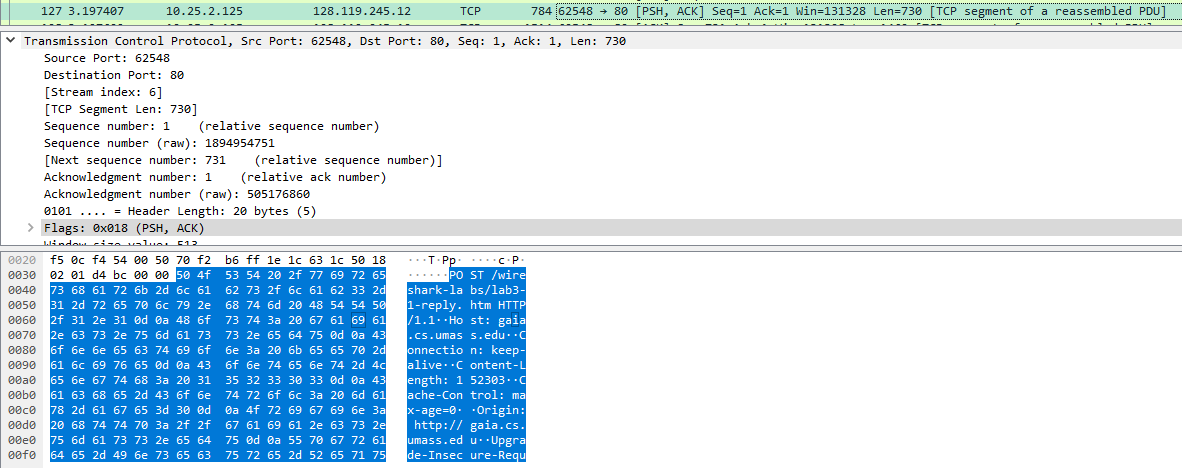


Answer: from figure #3, Sequence number: 0, Acknowledgment number: 1, Initial sequence number +1,

Flag: 0x012( SYN, ACK)

6. What is the sequence number of the TCP segment containing the HTTP POST command? Note that in order to find the POST command, you’ll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a “POST” within its DATA field.

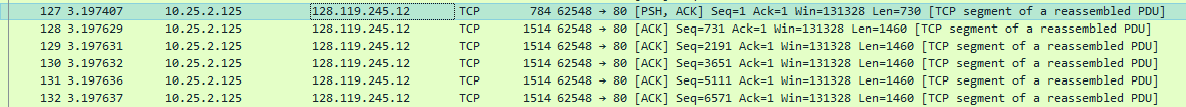
Figure #4



Answer: from figure #4, sequence number: 1

7. Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST)? At what time was each segment sent? When was the ACK for each segment received? Given the difference between when each TCP segment was sent, and when its acknowledgement was received, what is the RTT value for each of the six segments? What is the EstimatedRTT value (see Section 3.5.3, page 239 in text) after the receipt of each ACK? Assume that the value of the EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the EstimatedRTT equation on page 239 for all subsequent segments.

Figure #5



Answer: from figure #5, Sequence numbers: 1, 731, 2191, 3651, 5111, 6571

Time each segment sent: 3.197407, 3.197629, 3.197631, 3.197632, 3.197636, 3.197637

Time ACK for each segment received: 3.210314, 3.383281, 3.384168, 3.384169, 3.566954, 3.570233

RTT values: 0.012907, 0.185652, 0.186537, 0.186537, 0.369318, 0.372596

EstimatedRTT = 0.875 \* EstimatedRTT + 0.125 \* SampleRTT

EstimatedRTT:

1 -> 0.012907

2 ->0.875×0.012907+0.125× 0.185652= 0.0345

3 ->0.875×0.0345+0.125×0.186537= 0.0535

4 ->0.875×0.0535+0.125×0.186537= 0.0701

5 ->0.875×0.0701+0.125×0.369318= 0.1075

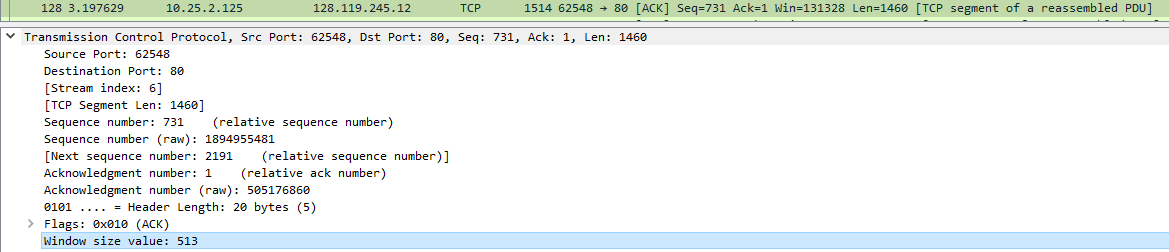
6 ->0.875×0.1075+0.125×0.372596= 0.140637

8. What is the length of each of the first six TCP segments?

Answer: from figure#5, 730, 1460, 1460, 1460, 1460, 1460

9. What is the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?

Figure #6



Answer: from figure #6, 513 bytes and the lack of receiver buffer space never throttle the sender

10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?

Answer: No retransmitted segments. Sequence number of each TCP is always increasing

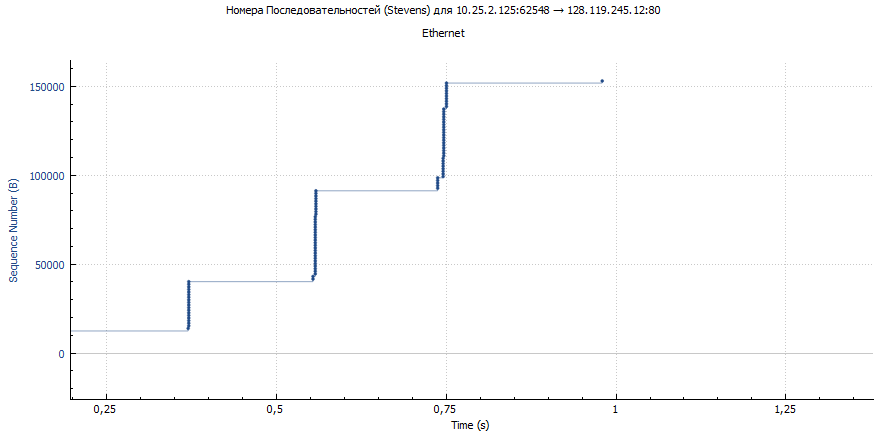
11. How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment

Answer: The receiver typically acknowledge amount of data equal to the difference between next two acknowledged sequence numbers

12. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.

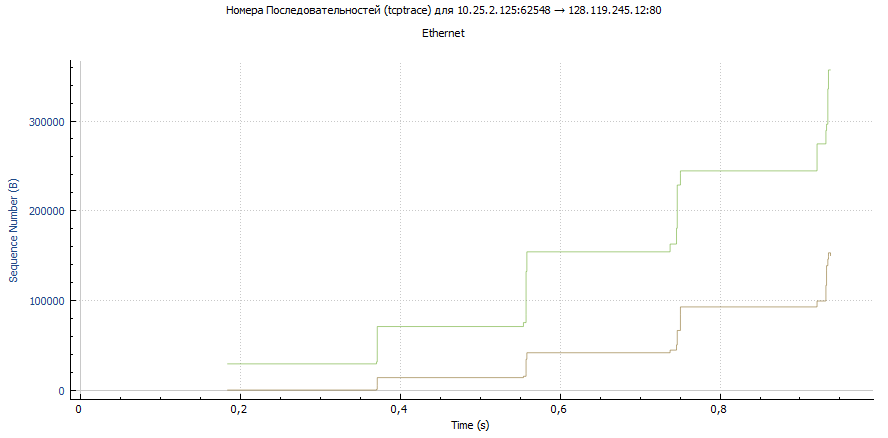
Answer: Throughput is equal to File\_Weight divided by time where time is Last ACK time minus time of CP segment with HTTP POST. Thus, throughput = 155648 bytes / ( 3.763356sec – 3.197407sec ) = 275024.1632 bytes/second

13. Use the Time-Sequence-Graph(Stevens) plotting tool to view the sequence number versus time plot of segments being sent from the client to the gaia.cs.umass.edu server. Can you identify where TCP’s slowstart phase begins and ends, and where congestion avoidance takes over? Comment on ways in which the measured data differs from the idealized behavior of TCP that we’ve studied in the text.



Answer: From 0.01 to 0.32

14. Answer each of two questions above for the trace that you have gathered when you transferred a file from your computer to gaia.cs.umass.edu



Answer: from 0.95 to 0.38